

BEST AVAILABLE COPY

Japanese Unexamined Patent Application Publication

No. 9-146835

[Title of the Invention] SYSTEM FOR DATA PREFETCHING

[Abstract]

[Object] The invention has an object to reduce, in a microprocessor system, the waiting time for fetching data from a memory.

[Solving Means] In a data processing system primary and secondary caches, a stream filter and a buffer, prefetching of cache lines is inventively executed. In a mode 1, data are not prefetched. In a mode 2, two cache lines are prefetched: a line is prefetched to L1 cache, and the following line is prefetched to a stream buffer. In mode 3, three or more cache lines are prefetched at a time. Prefetching can be executed upon occurrence of a cache error or hit. Upon occurrence of a cache error in a continuous cache line, the stream of the cache line can be assigned to a stream buffer.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-146835

(43) 公開日 平成9年(1997) 6月6日

(51) Int.Cl. ⁹	識別記号	序内整理番号	F I	技術表示箇所
G 0 6 F 12/08		7623-5B	G 0 6 F 12/08	D
		7623-5B		C
	3 1 0	7623-5B		3 1 0 Z

審査請求 未請求 請求項の数20 O L (全 22 頁)

(21) 出願番号 特願平8-225449

(22) 出願日 平成8年(1996) 8月27日

(31) 優先権主張番号 08/554180

(32) 優先日 1995年11月6日

(33) 優先権主張国 米国 (U S)

(71) 出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション

INTERNATIONAL BUSINESS MACHINES CORPORATION

アメリカ合衆国10504、ニューヨーク州
アーモンク (番地なし)(72) 発明者 マイケル・ジェイ・メイフィールド
アメリカ合衆国78730、テキサス州オース
ティン、グレンレイク・ドライブ 911

(74) 代理人 弁理士 合田 潔 (外2名)

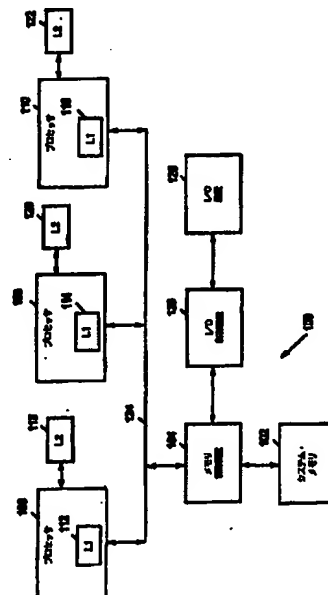
最終頁に続く

(54) 【発明の名称】 データのプリフェッチのためのシステム

(57) 【要約】

【課題】 マイクロプロセッサ・システムにおいて、メモリからのデータのフェッチのための待ち時間を低減する。

【解決手段】 1次及び2次キャッシュ、並びにストリーム・フィルタ及びバッファを実現するデータ処理システムにおいて、キャッシュ・ラインのプリフェッチが進歩的に実行される。第1のモードでは、データがプリフェッチされない。第2のモードでは、2つのキャッシュ・ラインがプリフェッチされ、1ラインは1キャッシュに、次のラインはストリーム・バッファにプリフェッチされる。第3のモードでは、3つ以上のキャッシュ・ラインが1度にプリフェッチされる。プリフェッチはキャッシュ・ミスまたはヒットに対して実行され得る。連続キャッシュ・ラインにおけるキャッシュ・ミスは、キャッシュ・ラインのストリームをストリーム・バッファに割当て得る。



【特許請求の範囲】

【請求項1】 プロセッサと、

前記プロセッサにバスを介して接続されるシステム・メモリと、

前記プロセッサに接続される第1のキャッシュと、

前記プロセッサに接続される第2のキャッシュと、

前記システム・メモリからプリフェッチされる1つ以上のデータ・ラインを記憶するストリーム・バッファ回路と、

プリフェッチ状態を示すストリーム・フィルタ回路と、

前記ストリーム・フィルタ回路に接続され、前記システム・メモリから前記第1及び第2のキャッシュ、並びに前記ストリーム・バッファ回路へのデータのフェッチ及びプリフェッチを選択的に制御する制御回路と、

を含む、データ処理システム。

【請求項2】 前記制御回路が、

第1のキャッシュ・ラインに対する要求を前記プロセッサから受信する回路と、

前記第1のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断する回路と、

前記第1のキャッシュ・ラインが前記第1のキャッシュ内に存在せず、前記第1のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第1のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチする回路と、

前記第1のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第1のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにフェッチする回路と、

前記第1のキャッシュ・ラインのアドレスを増分し、第1の増分アドレスを生成する回路と、

前記増分アドレスを前記ストリーム・フィルタに記憶する回路と、

を含む、請求項1記載のシステム。

【請求項3】 前記制御回路が、

前記増分アドレスを有する第2のキャッシュ・ラインに対する要求を前記プロセッサから受信する回路と、

前記第2のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断する回路と、

前記増分アドレスが前記ストリーム・フィルタ内に存在するか否かを判断する回路と、

前記増分アドレスを増分し、第2の増分アドレスを生成する回路と、

前記第2の増分アドレスで開始するストリームを割当てる回路と、

前記第2の増分アドレスを有する第3のキャッシュ・ラインが、前記第1のキャッシュ内に存在するか否かを判断する回路と、

前記第2のキャッシュ・ラインが前記第2のキャッシュ内に存在し、前記第2のキャッシュ・ラインが前記第1

のキャッシュ内に存在しない場合、前記第2のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチする回路と、

前記第2のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第2のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにフェッチする回路と、

前記第3のキャッシュ・ラインが前記第2のキャッシュ内に存在し、前記第3のキャッシュ・ラインが前記第1のキャッシュ内に存在しない場合、前記第3のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにプリフェッチする回路と、

前記第3のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第3のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにプリフェッチする回路と、

を含む、請求項2記載のシステム。

【請求項4】 前記第2の増分アドレスを増分し、第3の増分アドレスを生成する回路と、

前記第3の増分アドレスを有する第4のキャッシュ・ラインが、前記第2のキャッシュ内に存在するか否かを判断する回路と、

前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第4のキャッシュ・ラインを前記システム・メモリから前記ストリーム・バッファ回路にプリフェッチする回路と、

前記第3のキャッシュ・ラインに対する要求を前記プロセッサから受信する回路と、

前記第3のキャッシュ・ラインを前記第1のキャッシュから前記プロセッサに送信する回路と、

前記第4のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断する回路と、

前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第4のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチする回路と、

前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第4のキャッシュ・ラインを前記ストリーム・バッファ回路から前記第1のキャッシュにフェッチする回路と、

前記第3の増分アドレスを増分し、第4の増分アドレスを生成する回路と、

前記第4の増分アドレスを有する第5のキャッシュ・ラインが、前記第2のキャッシュ内に存在するか否かを判断する回路と、

前記第5のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第5のキャッシュ・ラインを前記第2のキャッシュから前記ストリーム・バッファ回路にフェッチする回路と、

前記第5のキャッシュ・ラインが前記第2のキャッシュ

内に存在しない場合、前記第5のキャッシュ・ラインを前記システム・メモリから前記ストリーム・バッファ回路にプリフェッチする回路と、を含む、請求項3記載のシステム。

【請求項5】前記ストリーム・バッファ回路に記憶される1つ以上のキャッシュ・ラインが、前記第1のキャッシュ内に配置される、請求項1記載のシステム。

【請求項6】前記ストリーム・バッファ回路に記憶される1つ以上のキャッシュ・ラインが、前記プロセッサを含むチップ内に配置される、請求項1記載のシステム。

【請求項7】前記ストリーム・バッファ回路に記憶される1つ以上のキャッシュ・ラインが、前記プロセッサ及び前記システム・メモリに接続されるノード制御装置内に配置される、請求項1記載のシステム。

【請求項8】前記第1のキャッシュが前記プロセッサと同一チップ上に配置される1次キャッシュであり、前記第2のキャッシュが前記チップの外部に配置される2次キャッシュである、請求項1記載のシステム。

【請求項9】前記ストリーム・フィルタ回路が複数のストリームを追跡可能であり、前記ストリーム・フィルタ回路内の各エントリが、前記複数のストリームの1つを追跡し、前記各エントリが、当該エントリにより追跡される前記ストリームの妥当性を示す第1の標識と、当該エントリのアドレスの増分方向を示す第2の標識とを含む、請求項1記載のシステム。

【請求項10】前記ストリーム・バッファ回路が各エントリに対して、1) ページ・アドレスと、2) ライン・アドレスと、3) 妥当性標識とを含む、請求項1記載のシステム。

【請求項11】前記ストリーム・フィルタ回路内のエントリが実アドレスを有し、前記ストリーム・バッファ回路内のエントリが有効アドレスを有する、請求項1記載のシステム。

【請求項12】前記プロセッサ、前記制御回路、前記ストリーム・フィルタ回路、前記第1のキャッシュ、及び前記ストリーム・バッファ回路の一部が、同一チップ上に配置される、請求項1記載のシステム。

【請求項13】第2のプロセッサと、前記第2のプロセッサに接続される前記システム・メモリと、前記第2のプロセッサに接続される第3のキャッシュと、前記第2のプロセッサに接続される第4のキャッシュと、前記第2のプロセッサに接続され、前記システム・メモリからプリフェッチされる1つ以上のデータ・ラインを記憶する第2のストリーム・バッファ回路と、前記第2のプロセッサに接続され、プリフェッチ状態を示す第2のストリーム・フィルタ回路と、前記第2のストリーム・フィルタ回路に接続され、前記

システム・メモリから前記第3及び第4のキャッシュ、並びに前記第2のストリーム・バッファ回路へのデータのフェッチ及びプリフェッチを選択的に制御する、第2の制御回路と、

を含む、請求項1記載のシステム。

【請求項14】データ処理システムにおいて、

第1のキャッシュ・ラインに対する要求を前記プロセッサから受信するステップと、

前記第1のキャッシュ・ラインが第1のキャッシュ内に存在するか否かを判断するステップと、

前記第1のキャッシュ・ラインが前記第1のキャッシュ内に存在せず、前記第1のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第1のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチするステップと、

前記第1のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第1のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにフェッチするステップと、

前記第1のキャッシュ・ラインのアドレスを増分し、第1の増分アドレスを生成するステップと、

前記増分アドレスをストリーム・フィルタに記憶するステップと、

を含む、方法。

【請求項15】前記増分アドレスを有する第2のキャッシュ・ラインに対する要求を前記プロセッサから受信するステップと、

前記第2のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断するステップと、

前記増分アドレスが前記ストリーム・フィルタ内に存在するか否かを判断するステップと、

前記増分アドレスを増分し、第2の増分アドレスを生成するステップと、

前記第2の増分アドレスで開始するストリームを割当てるステップと、

前記第2の増分アドレスを有する第3のキャッシュ・ラインが、前記第1のキャッシュ内に存在するか否かを判断するステップと、

前記第2のキャッシュ・ラインが前記第2のキャッシュ内に存在し、前記第2のキャッシュ・ラインが前記第1のキャッシュ内に存在しない場合、前記第2のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチするステップと、

前記第2のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第2のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにフェッチするステップと、

前記第3のキャッシュ・ラインが前記第2のキャッシュ内に存在し、前記第3のキャッシュ・ラインが前記第1のキャッシュ内に存在しない場合、前記第3のキャッシ

ユ・ラインを前記第2のキャッシュから前記第1のキャッシュにプリフェッチするステップと、
前記第3のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第3のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにプリフェッチするステップと、
を含む、請求項14記載の方法。

【請求項16】前記第2の増分アドレスを増分し、第3の増分アドレスを生成するステップと、
前記第3の増分アドレスを有する第4のキャッシュ・ラインが、前記第2のキャッシュ内に存在するか否かを判断するステップと、

前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第4のキャッシュ・ラインを前記システム・メモリから前記ストリーム・バッファ回路にプリフェッチするステップと、

前記第3のキャッシュ・ラインに対する要求を前記プロセッサから受信するステップと、

前記第3のキャッシュ・ラインを前記第1のキャッシュから前記プロセッサに送信するステップと、

前記第4のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断するステップと、

前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第4のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチするステップと、

前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第4のキャッシュ・ラインを前記ストリーム・バッファ回路から前記第1のキャッシュにフェッチするステップと、

前記第3の増分アドレスを増分し、第4の増分アドレスを生成するステップと、

前記第4の増分アドレスを有する第5のキャッシュ・ラインが、前記第2のキャッシュ内に存在するか否かを判断するステップと、

前記第5のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第5のキャッシュ・ラインを前記第2のキャッシュから前記ストリーム・バッファ回路にフェッチするステップと、

前記第5のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第5のキャッシュ・ラインを前記システム・メモリから前記ストリーム・バッファ回路にプリフェッチするステップと、

を含む、請求項15記載の方法。

【請求項17】前記ストリーム・バッファ回路に記憶される1つ以上のキャッシュ・ラインが、前記第1のキャッシュ内に配置される、請求項16記載の方法。

【請求項18】前記第1のキャッシュが前記プロセッサと同一チップ上に配置される1次キャッシュであり、前記第2のキャッシュが前記チップの外部に配置される2

次キャッシュである、請求項17記載の方法。

【請求項19】前記ストリーム・フィルタ回路が複数のストリームを追跡可能であり、前記ストリーム・フィルタ回路内の各エントリが、前記複数のストリームの1つを追跡し、前記各エントリが、当該エントリにより追跡される前記ストリームの妥当性を示す第1の標識と、当該エントリのアドレスの増分方向を示す第2の標識とを含む、請求項18記載の方法。

【請求項20】前記ストリーム・フィルタ回路内のエントリが実アドレスを有し、前記ストリーム・バッファ回路内のエントリが有効アドレスを有する、請求項19記載の方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は一般にデータ処理システムに関し、特に、データをメモリからプリフェッチするシステムに関する。

【0002】

【従来の技術】現プログラム及びデータをプロセッサ（CPU）が高速に使用可能にすることにより、データ処理システムの処理速度を向上するために、時に特殊な超高速メモリが使用される。こうした高速メモリはキャッシュとして知られ、時に大規模コンピュータ・システムにおいて、主メモリ・アクセス時間とプロセッサ論理回路間の速度差を補償するために使用される。プロセッサ論理回路は、通常主メモリ・アクセス時間よりも高速であり、処理速度は大抵主メモリの速度により制限される。動作速度の不一致を補償する1つの技法は、CPUと主メモリとの間で、そのアクセス時間がプロセッサ論理回路の伝播遅延に近い超高速の小メモリを使用する。これは現在CPU内で実行されているプログラムのセグメント、及び現計算において頻繁に必要とされる一時データを記憶するために使用される。プログラム（命令）及びデータを高速に使用可能にすることにより、プロセッサの性能を高めることが可能である。

【0003】通常の多数のプログラムの分析の結果、所与の時間間隔におけるメモリ参照は、メモリ内の幾つかのローカル化領域に制限されることが判明した。この現象は“参照のローカル性”の特性として知られる。この特性の理由は、通常のコンピュータ・プログラム・フローが直線的に流れ、しばしばプログラム・ループ及びサブルーチン呼び出しに遭遇することを考慮することにより理解され得る。プログラム・ループが実行されるとき、CPUはループを構成するメモリ内の命令セットを繰り返し参照する。所与のサブルーチンが呼び出される度に、その命令セットがメモリからフェッチされる。従って、ループ及びサブルーチンは、命令フェッチのためのメモリ参照をローカル化する傾向がある。データのメモリ参照も、より低い度合いでローカル化される傾向がある。テーブル・ルックアップ・プロシジャは、テーブル

が記憶されるメモリ部分を繰り返し参照する。反復プロセスは共通メモリ位置を参照し、数字のアレイがメモリの局所部分に記憶される。これらの観測の結果が参照のローカル性であり、これは短い時間間隔において、通常のプログラムにより生成される命令のアドレスが、メモリの幾つかのローカル化領域を繰り返し参照し、残りのメモリ領域は比較的稀にアクセスされることを示す。

【0004】プログラム及びデータのアクティブ部分が高速の小メモリ内に配置される場合、平均メモリ・アクセス時間が低減され、従ってプログラムの総実行時間が減少する。こうした高速小メモリは、前述のように、キャッシュとして参照される。キャッシュ・メモリ・アクセス時間は、主メモリのアクセス時間よりも、しばしば5倍乃至10倍小さい。キャッシュはメモリ階層において最も高速のコンポーネントであり、CPUコンポーネントの速度に近い。

【0005】キャッシュ構成の基本的な思想は、最も頻繁にアクセスされる命令及びデータを高速キャッシュ・メモリに保存することにより、平均メモリ・アクセス時間をキャッシュのアクセス時間に近づけることである。キャッシュは主メモリのサイズの小部分ではあるが、メモリ要求の大部分は、プログラムの参照のローカル性のために、高速キャッシュ・メモリ内で見い出される。

【0006】キャッシュの基本動作は次のようである。CPUがメモリ・アクセスを必要とすると、キャッシュが調査される。ワードがキャッシュ内で見い出されると、それは高速メモリから読出される。CPUによりアドレス指定されるワードがキャッシュ内で見い出されない場合には、そのワードを読出すために、主メモリがアクセスされる。アクセスされたばかりのワードを含むワード・ブロックが、次に主メモリからキャッシュ・メモリに転送される。このようにして、特定のデータがキャッシュに転送され、将来のメモリ参照が、要求されるワードを高速キャッシュ・メモリ内で見い出すことになる。

【0007】コンピュータ・システムの平均メモリ・アクセス時間は、キャッシュの使用により多大に改良され得る。キャッシュ・メモリの性能は、しばしば“ヒット率”と呼ばれる量により測定される。CPUがメモリを参照し、ワードをキャッシュ内で見い出すとき、これは“ヒット”を生成したと称される。ワードがキャッシュ内で見い出されない場合、そのワードは主メモリ内に存在し、“ミス”としてカウントされる。ヒット率が十分に高く、ほとんどの時間、CPUが主メモリの代わりにキャッシュをアクセスする場合、平均アクセス時間は高速キャッシュ・メモリのアクセス時間に近づく。例えば、100nsのキャッシュ・アクセス時間、1000nsの主メモリ・アクセス時間、及び0.9のヒット率を有するコンピュータでは、200nsの平均アクセス時間を生成する。これはキャッシュ・メモリを有さず、その

アクセス時間が1000nsの類似のコンピュータに比較して、多大な改良である。

【0008】最新のマイクロプロセッサでは、プロセッサ・サイクル時間が技術の進歩と共に、向上し続けている。また、思惑実行、深いパイプライン、並びにより多くの実行要素などの設計技法が、マイクロプロセッサの性能を改良し続けている。性能の改良はメモリ・インタフェースに、より大きな負担を課すことになる。なぜなら、プロセッサが、より多くのデータ及び命令を、メモリからマイクロプロセッサに供給することを要求するからである。メモリ待ち時間を低減するために、大規模なオンチップ・キャッシュ（1次またはL1キャッシュ）が実現され、これらはしばしば、より大規模なオフチップ・キャッシュ（2次またはL2キャッシュ）により増補される。

【0009】メモリ・データを予め1次キャッシュに供給し、待ち時間を低減するために、しばしばプリフェッチ技法が実現される。理想的には、プログラムは、プロセッサがメモリ・データを必要とするとき、そのコピーが常に1次キャッシュ内に存在するように、データ及び命令を予め十分にプリフェッチする。

【0010】問題は、マイクロプロセッサ・アーキテクチャが、全ての場合において必要とされ得るデータのアドレスを明示的に決定するための、十分な事前情報を提供しないことである。例えば、メモリ内におけるデータ・オペランドのアドレスは、それ自身メモリ内にあり、メモリ命令により使用される第1命令によりフェッチされなければならない。こうしたシーケンスでは、プロセッサはプリフェッチを実行するためのアドレスを前もって有さない。

【0011】命令及び（または）データのプリフェッチは既知である。しかしながら、既存のプリフェッチ技法は、しばしば命令及び（または）データを早計にプリフェッチする。命令及び（または）データをプリフェッチし、それらを使用しないことに関わる問題は、（1）プリフェッチされたデータが、プロセッサにより必要とされるデータを置換し得る、（2）プリフェッチ・メモリ・アクセスが、続くプロセッサ・キャッシュ・リロードを生成し、それによりプリフェッチ・アクセスが待機し、必要データの待ち時間を増加し得ることである。これら両方の作用はCPUの効率を低下させる。従って、マイクロプロセッサの性能を低下すること無く、キャッシュ・ミスによる1次キャッシュへのデータ及び命令アクセスの待ち時間を低減する、プリフェッチのための改良されたシステムが必要とされる。

【0012】

【発明が解決しようとする課題】本発明の目的は、マイクロプロセッサにおいて、1次及び2次キャッシュと共にストリーム・フィルタを使用して、メモリからプリフェッチ・データを提供し、マイクロプロセッサ・システ

10

20

30

40

50

ムにおけるデータ待ち時間を低減することである。

【0013】本発明の別の目的は、複数のストリームを同時にサポートでき、プリフェッチ・データをしだいに増分して、プリフェッチの深さを制御できる固有のストリーム・フィルタ装置を提供することである。

【0014】

【課題を解決するための手段】前記記述は、以降で述べられる本発明の詳細がより理解されるように、本発明の特長及び技術的利点をかなり広く述べたものであり、本発明の追加の特長及び利点については、以降で詳細に述べることにする。

【0015】

【発明の実施の形態】以降の説明では、本発明を十分に理解するために、例えば特定のワード長またはバイト長などの、多数の特定の詳細が述べられる。しかしながら、当業者には明らかなように、本発明はこうした特定の詳細無しでも実施し得る。他の例では、本発明を不要な詳細により不明瞭化しないように、既知の回路がブロック図形式で示される。大部分において、タイミングなどに関する詳細は、本発明を理解する上で不要であり、当業者の知る範囲内である限り省略される。

【0016】図面を通じて、図示の要素は必ずしもその縮尺は適正ではなく、同一または類似の要素は同一の参照番号により示される。

【0017】図1を参照して、本発明を有利に実現するデータ処理システム100について説明する。マルチプロセッサ・システム100は、システム・バス124に動作上接続される複数の処理ユニット106、108、110を含む。任意の数の処理ユニットが、システム100内において使用され得る。システム・バス124には更に、システム・メモリ102へのアクセスを制御するメモリ制御装置104が接続される。メモリ制御装置104はI/O制御装置126にも接続され、I/O制御装置126はI/O装置128に接続される。処理ユニット106、108、110、I/O制御装置126、及びI/O装置128は、ここでは全てバス装置として参照される。図示のように、各処理ユニット106、108、110は、プロセッサ及びL1(1次)キャッシュ112、114、116をそれぞれ含み得る。L1キャッシュは、それぞれのプロセッサと同一チップ上に配置され得る。処理ユニット106、108、110には、それぞれL2(2次)キャッシュ118、120、122が接続される。これらのL2キャッシュは、それぞれが接続されるプロセッサを介して、システム・バス124に接続される。

【0018】各L1及びL2キャッシュ対は通常、直列に関連付けられる。L1キャッシュはストア・イン(store-in)またはライト・スルー(write-through)として実現され、より大規模で低速のL2キャッシュは、ライト・バック(write-back)・キャッシュとして実現さ

れ得る。L1及びL2キャッシュ制御装置(図示せず)の両方は、処理ユニットの一部として物理的に実現され、内部的にバスを介して処理ユニットに接続される。L2キャッシュ制御装置はオフ・チップであってもよい。

【0019】図2を参照すると、本発明に従い動作するように構成され得るデータ処理システム200が示される。システム200は、システム100の代替アーキテクチャである。システム100及びシステム200内において、プロセッサ及びキャッシュの基本動作は類似である。メモリ制御装置104及びノード制御装置205の制御及び機能は、本発明に関して類似である。

【0020】システム200において、プロセッサ201は内部L1キャッシュ202を有し、L1キャッシュ202は外部L2キャッシュ203に接続される。プロセッサ201は、バス204によりノード制御装置205に接続される。ノード制御装置205は、プロセッサ201をシステム200の残りの要素に接続するための既知の基本機能を実行する。ノード制御装置205はバス206により、スイッチ207に接続される。スイッチ207は例えばクロス・ポイント・スイッチであり、他のプロセッサ及び(または)I/O装置(図示せず)を、バス208を介してシステム・メモリ209に接続する。以降の議論はシステム200に関して述べられるが、後述の本発明に関する議論はシステム100にも関連付けられ、またそこでも実現され得る。

【0021】本発明の目標は、プロセッサ201がキャッシュ・ラインをL1キャッシュ202内で相当な時間ヒットするように、これらのキャッシュ・ライン(データ部分またはデータ・ブロック)をL1キャッシュ202にプリフェッチするための、効率的で的確な技法を提供することであり、それにより、プロセッサ201の性能を低下させるシステム・メモリ209からのアドレス及びデータ情報の取り出しを最小化する。

【0022】図3を参照すると、使用されないデータのプリフェッチの発生を低減するために、ストリーム・フィルタが使用される。これらのストリーム・フィルタは、アドレス及び方向情報を含む履歴バッファである。ストリーム・フィルタは、L1キャッシュ内でミスしたキャッシュ・ラインの次に順次的に高いキャッシュ・ラインのアドレスを含む。プロセッサからのアクセスが、次に高いキャッシュ・ラインに対して実施されると、ストリーム状態が検出され、ストリーム・バッファが割当てられる。ストリーム・フィルタは、ライン・アドレス"X"へのアクセスが発生した場合、ライン・アドレス"X+1"を書込まれる。"X+1"がストリーム・フィルタ内に存在する間に、続くアクセスがアドレス"X+1"に対して実施されると(フィルタ・ヒット)、"X+1"がストリームとして割当てられる。

【0023】ストリーム・バッファは、潜在的キャッシ

10

20

30

40

50

ユ・データを保持するプリフェッチ・バッファである。その思想は、プロセッサ内で実行されるプログラムがデータ/命令の順次ストリームを実行している場合、追加のラインをシステム・メモリからストリーム・バッファにプリフェッチすることが有用であり得ることによる。従って、続くキャッシュ・ミスがデータをストリーム・バッファ内で見い出すことができ、ストリーム・バッファがデータをL1キャッシュ及び(または)プロセッサに供給する。

【0024】ストリーム・フィルタ及びストリーム・バッファは、ストリーム・バッファをもミスするL1キャッシュ・ミスが発生する場合に、ミス・アドレスがストリーム・フィルタ内に記憶されるアドレスと比較されるように協働する。ストリーム・フィルタ内でヒットが発生する場合(順次データ・ラインへの順次アクセスが存在したことを意味する)、次のラインもまた将来的に必要とされる可能性が高い。

【0025】図4を参照すると、本発明に従い構成されるシステム200の詳細図、並びにCPU201内のデータ・フローが示される。データ・フローの変形も既知であり、そうしたものには、例えば命令及びデータ用の別々のL1キャッシュの使用などが含まれる。L1キャッシュ202は、任意の既知の置換ポリシーにより、頻繁に使用されるデータのコピーをメモリ209から保持する。大規模なL2キャッシュ203は、L1キャッシュ202よりも多くのデータを保持し、メモリ・コピーレンス・プロトコルを普通に制御する。L1キャッシュ202内のデータは、L2キャッシュ203内のデータのサブセットであり得る。L1キャッシュ202及びL2キャッシュ203は、“ストア・イン”・キャッシュである。他の機能要素(I/Oを含む)は、既知のスヌープ・プロトコルを用いて、データを競合する。スヌーピングの1つの形態が、米国特許出願番号第442740号で開示されている。

【0026】CPU201に対して示される境界は、チップ境界及び機能境界を示すが、これらは本発明の範囲を制限するものではない。PCC404はプロセッサ・キャッシュ制御装置であり、メモリ・サブシステムからのフェッチ(プリフェッチ)及びストアを制御する。PCC404は、L1キャッシュ202に対するディレクトリ410の作成、有効アドレスから実アドレスへの変換、及びその逆の変換などの、他の既知の機能を有する。プリフェッチ・バッファ(PBFR)402は、CPU201及びL1キャッシュ202にステージされる数ラインのメモリ・データを保持する。PBFR402はストリーム・バッファである。

【0027】PCC404がデータをフェッチするとき、そのデータがL1キャッシュ202内に存在すれば(L1ヒット)、データがPCC404に送信される。L1キャッシュ202内に存在しないが(L1ミス)、

L2キャッシュ203内に存在する場合(L2ヒット)、L1キャッシュ202内のあるラインが、L2キャッシュ203からのこの主体データにより置換される。この場合、データはL1キャッシュ202及びPCC404に同時に送信される。L2キャッシュ203においても同様にミスが発生する場合には、データはメモリ209からBIU401にフェッチされ、L1キャッシュ202、L2キャッシュ203、及びPCC404に同時にロードされる。このオペレーションの変形は既知である。データ・ストア・オペレーションは、データがL1ラインに記憶されてオペレーションが完了する以外は、フェッチ・オペレーションと類似である。

【0028】以降の議論では、ストリーム・バッファ(図3参照)の様々な部分が、システム200の様々な部分に配置される。本技法では、ストリーム・バッファは4つのキャッシュ・ラインを記憶する能力を有するが、任意の数のキャッシュ・ラインがストリーム・バッファ内で実現され得る。ストリーム・バッファの1キャッシュ・ラインは、L1キャッシュ202内で実現される。本来、L1キャッシュ202内のキャッシュ・ラインの1つは、ストリーム・バッファのキャッシュ・ラインの内の1つの機能として利用される。ストリーム・バッファの第2のキャッシュ・ラインは、PBFR(プリフェッチ・バッファ)402内に配置される。ストリーム・バッファの他の2つのキャッシュ・ラインは、ノード制御装置205内のPBFR2 405及びPBFR3 406に配置される。ノード制御装置205は、バス204に沿って、CPU201の下流のチップ上に配置され得る。システム100のアーキテクチャが使用される場合には、メモリ制御装置104がこれらのストリーム・バッファ・ラインを含み得る。

【0029】前記参照されたIEEE条項内で述べられるように、ストリーム・フィルタ及びストリーム・バッファの基本動作は、要求キャッシュ・ラインに対してL1キャッシュ・ミスが発生するとき、キャッシュ・ラインのアドレスが増分され(一般に1ライン・アドレス分)、この増分アドレスがストリーム・フィルタ403に挿入される。L1キャッシュ202内におけるキャッシュ・ラインの続くミスの発生に際して、このL1キャッシュ・ミスのアドレスが、ストリーム・フィルタ403に含まれるアドレスと比較される。少なくとも1つのアドレスとの一致が観測されると、キャッシュ・ラインのストリームがストリーム・バッファ内に割当てられる。

【0030】上述のように、キャッシュ・ミスが発生するとき、ストリーム・フィルタ・バッファには次の順次キャッシュ・ラインのアドレスが書込まれる。ストリーム・フィルタ(図3及び図5参照)は、こうした事象の“履歴”を含むアドレスを保持できる複数の位置を含む。これらは最低使用頻度(LRU)ベースで置換され

得る。キャッシュ・ミスが発生する都度、ストリーム・フィルタ内のアドレスが、キャッシュ・ライン・ミスのアドレスと比較される。ヒットが発生する場合、フィルタ・ヒットが発生したと称され、ストリームが割当てられる。ストリーム・モードでは、余分なキャッシュ・ラインが、L1キャッシュ202によりストリームの一部として必要とされるとの期待から、ストリーム・バッファ(例えば、L1キャッシュ202、PBFR402、PBFR2 405、PBFR3 406内のライン)にプリフェッチされる。

【0031】図5は、本発明によるストリーム・フィルタ及びストリーム・バッファの動作を示す高レベル機能図を示す。CPU201は、使用されるアーキテクチャに従い、有効アドレス(EA)を生成する。EAは潜在的にオフセットを有する要求データ・アドレスである。変換器503により、CPU201は、EAに対応する変換アドレスまたは実アドレス(RA)を生成する。実アドレスはフィルタ・キュー502により使用される。が、フィルタ・キュー502が実アドレスの代わりに有効アドレスを使用することも、本発明の範囲に含まれる。比較器504により、RAはフィルタ・キュー502内のRAと無差別に比較され、エントリがその有効ビット(V)により有効と示される場合には、一致はフィルタ・ヒットと呼ばれる。

【0032】フィルタ・キュー502は、各エントリに対して推測方向標識を含み、これは推測ストリームが増分または減分(±1またはアップ/ダウン)されるべきことを示す。これについては後述する。

【0033】各フィルタ・キュー・エントリは、そのアドレスに対応するストリームが存在するか否かを示すフィールドを含み、存在する場合、そのストリームの識別ストリーム番号を含む(1度に複数のストリームが割当てられ得る)。

【0034】フィルタ・ヒットが発生するとき、ストリームがストリーム・アドレス・バッファ501に割当てられ、対応する割当てがストリーム・データ・バッファ506に作成される。ストリーム・アドレス・エントリは、特定の割当てストリームの次のデータ・ラインの推測有効アドレスを含む。再度、ここでも有効アドレスの代わりに、実アドレスを使用することも可能である。ストリーム・アドレス・エントリは更に、ストリームが割当てられることを示す有効ビットを含む。更に、ストリームの状態を追跡するために使用される状態フィールドが存在する。また、推測方向(±またはアップ/ダウン)のコピーが、ストリーム・バッファ内に保持される。比較器505はプロセッサにより発行されるEAを、ストリーム・アドレス・バッファ501に含まれるページ・アドレス及びライン・アドレスと比較する。一致が発生する場合、これはストリーム・ヒットと呼ばれる。

【0035】図5に示される機能は別の方法によっても実現され、これらも本発明の範囲に含まれる。

【0036】システム・メモリ209内のメモリ空間は、128バイトのラインに分割され得る。各ラインは、ラインの偶数部分がアドレス0乃至63に、奇数部分がアドレス64乃至127に帰するように、半分に分割されてもよい。上述のように、CPU201は論理アドレス(EA)を生成し、これがメモリ内のキャッシュ可能なラインを指す実アドレスに変換される。メモリは2Nバイトのページに分割される。ページは、サイズ的にキャッシュ・エントリに対応するラインに分割される。キャッシュ・ミスの発生は、関連する実アドレスが分析される。実アドレスがラインの偶数部分に存在する場合、潜在ストリームは増分ストリームとなる。フィルタ・キュー502内のLRUフィルタ・キュー・エントリが、"アップ"方向によりマークされ、ライン・ミスRAが"1"増分されて、エントリに保管される。RAがラインの奇数側の場合には、キュー502内のRAエントリが1減分され、"ダウン"がエントリ内にマークされる。

【0037】別の技法として、ミスに際して、RAをフィルタ・エントリに保管し、続くミスをエントリと比較して、アップまたはダウンの方向を決定することも、本発明の範囲に含まれる。

【0038】ストリームが割当てられるとき、"次の"有効ライン・アドレスがストリーム・アドレス・バッファ501に保管されることが明らかであろう。バッファ501は各アクティブ・ストリームに対するエントリを含む。

【0039】L1キャッシュ202及びL2キャッシュ203のキャッシュ・ミスが発生する場合、システム・メモリ209をアクセスする以前に、ストリーム・バッファに対する問い合わせが生じる。フィルタ・キュー502回路及びストリーム・アドレス・バッファ501回路を結合することも、本発明の1つの態様である。

【0040】図6乃至図9を参照すると、本発明の進歩的プリフェッチ・モードのフロー図が示される。本発明は3つの進歩的プリフェッチ・モード及びそれらの変形を可能にする。それらは正常、データ・プリフェッチ、及びブラスト(Blast)である。正常モードでは、データはプリフェッチされない。データ・プリフェッチ・モードでは、2ラインがプリフェッチされ、1ラインがL1キャッシュに、1ラインがストリーム・バッファにプリフェッチされる。ブラスト・モードでは、3ライン以上が1度にプリフェッチされる。本発明の1つの態様では、ブラスト・モードにおいて、4ラインがプリフェッチされ、2ラインがデータ・プリフェッチ・モードでプリフェッチされ、追加の2ラインがストリーム・バッファにプリフェッチされる。任意のモードにおいて、プリフェッチ・バッファはプロセッサ・チップ、キャッシュ

・チップ、外部チップ、及び（または）メモリ・カード上に実装され得る。図6乃至図9は、ストリームが流れる推測方向が増分方向の場合の例を示す。減分方向の場合の例は、この例の変更として明らかであろう。図6乃至図9のフロー図は、データ・プリフェッチ・モード及びブラスト・モードに入る様子を示す。

【0041】ステップ601で、CPU201はキャッシュ・ラインAにおけるデータ・アクセスを開始する。ステップ602で、キャッシュ・ラインAがL1キャッシュ202内に存在するかどうかを判断する。存在する場合、プロセスはステップ603に移行し、キャッシュ・ラインAがCPU201に戻され、プロセスはステップ604で終了する。

【0042】しかしながら、キャッシュ・ラインAにおいてミスが発生すると、プロセスはステップ605に移行し、キャッシュ・ラインAのアドレスが、ストリーム・フィルタ403に含まれる全てのアドレスと比較される。

【0043】キャッシュ・ラインAがストリーム・フィルタ403内に存在しない場合、プロセスはステップ606に移行し、キャッシュ・ラインAのアドレスが1増分され、ストリーム・フィルタ403内に挿入される。従って、ステップ607で、キャッシュ・ラインAがL2キャッシュ203またはメモリ209から、L1キャッシュ202にフェッチされる。

【0044】図6乃至図9内のステップ607からステップ608への破線矢印は、ステップ608がステップ607の直後に、必ずしも発生しないことを示す。一般に多くのミスが発生するほど、キャッシュ・ラインA+1に対する要求（ステップ608）以前に、ストリーム・フィルタ403内のアドレス・エントリが発生し得る。

【0045】やがて、CPU201がキャッシュ・ラインA+1を要求し得る（ステップ608）。再度、PC404は、キャッシュ・ラインA+1がL1キャッシュ202内に存在するかどうかを判断する（ステップ609）。存在する場合、ステップ610でキャッシュ・ラインA+1がCPU201に返却され、プロセスはステップ611で終了する。すなわち、キャッシュ・ラインA+1がL1キャッシュ202内に存在するので、ストリーム・フィルタ403との比較は実行されず、A+1エントリはフィルタ置換アルゴリズムにより退却されるまで、フィルタ403内に留まる。フィルタ置換アルゴリズムは、係属中の米国特許出願番号第519032号における教示に従い実行され得る。しかしながら、L1キャッシュ202内でキャッシュ・ラインA+1に対するミスが発生すると、フィルタ・ヒットが発生し（ステップ637）、プロセスはステップ612に移行して、キャッシュ・ラインA+2で開始するキャッシュ・ライン・ストリームが割当てられる。なぜなら、要求キャッ

シュ・ラインA+1のアドレスが、ストリーム・フィルタ403内に存在するアドレスA+1と比較され、結果的にストリーム・フィルタ403内でヒットが生じるからである。次にステップ613で、キャッシュ・ラインA+1がL2キャッシュ203またはメモリ209から、L1キャッシュ202にフェッチされる。また、キャッシュ・ラインA+2が、L1キャッシュ202内に存在するかがチェックされる。存在しない場合、キャッシュ・ラインA+2がL2キャッシュ203またはメモリ209から、L1キャッシュ202にプリフェッチされる。

【0046】その後、ステップ614で、キャッシュ・ラインA+3がL2キャッシュ203内に存在するかが判断される。存在しない場合、プロセスはステップ615に移行し、キャッシュ・ラインA+3がメモリ209からプリフェッチされ、プリフェッチ・バッファPBFR402内に挿入される。しかしながら、キャッシュ・ラインA+3がL2キャッシュ203内に存在する場合には、プロセスはステップ615をスキップする。

【0047】再度、ステップ615からステップ616への破線矢印は、ステップ616がステップ615の直後に必ずしも発生しないことを示す。

【0048】ステップ616では、プロセッサ201はキャッシュ・ラインA+2を要求し得り、その場合、ラインA+2に対するアクセスが、L1キャッシュ202から実行される。ステップ613で、キャッシュ・ラインA+2がL1キャッシュ202にフェッチされたので、L1キャッシュ202はこのキャッシュ・ラインをCPU201に供給することができる。ステップ617で、アドレスA+3をストリームの先頭に有するよう、ストリーム・アドレス・バッファ501内のストリーム・アドレスが更新される。その後、ステップ618で、キャッシュ・ラインA+3がL1キャッシュ202内に存在するかがチェックされ、存在しない場合、キャッシュ・ラインA+3がL2キャッシュ203またはPBFR402から、L1キャッシュ202にフェッチされる。次にステップ619で、キャッシュ・ラインA+4がL2キャッシュ203またはメモリ209から、PBFR402にフェッチされる。

【0049】その後、システム200内でブラスト・モードが許可されていなければ（ステップ620）、プロセスはステップ616に戻り、CPU201が図示のように、キャッシュ・ラインを順次増分し続ける限り（“ストリーミング”として参照される）、ステップ616乃至621をループする。ステップ621は、その後ステップ616において、ラインA+3に対するL1キャッシュ・アクセスが発生し、ステップ617でストリームがアドレスA+3により更新され、ステップ618で、ラインA+4がL1キャッシュ202からフェッチされ、ステップ619で、キャッシュ・ラインA+4が

PBFR402からフェッチされることを示す。

【0050】上述の説明はデータ・プリフェッチ・モードを示すものである。ステップ620で、ブラスト・モードがシステム200において許可されている場合、プロセスはステップ622に移行し、CPU201からキャッシュ・ラインA+3に対する要求が発生する。ステップ622で、こうした要求に対して、PCC404がL1キャッシュ202内でキャッシュ・ラインA+3を探索する。キャッシュ・ラインA+3は、ステップ618によりL1キャッシュ202内に存在するので、キャッシュ・ラインA+3がCPU201に返却される。その後、ステップ623で、ストリーム・アドレス・バッファ501内のストリーム・アドレスが、A+4に更新される。ステップ624で、L1キャッシュ202にラインA+4が存在するか否かがテストされる。存在しない場合、キャッシュ・ラインA+4が、PBFR402から、L1キャッシュ202内のプリフェッチ・バッファ位置にフェッチされる。

【0051】その後、ステップ625で、キャッシュ・ラインA+5がL2キャッシュ203内に存在するか否かが判断される。存在する場合、プロセスはステップ626または627に移行する。本技法は、ノード制御装置205があらゆるストリーム・バッファ・アクセスを通知されることを要求する。通知が、次のストリーム・バッファがL2キャッシュ203内に存在せず、従ってフェッチされる必要がある場合に限定されると、ノード制御装置バッファ405及び406が一時的にプロセッサ201との同期を逸する。この設計のトレードオフの利点は、ステップ626及び627が結合され、ノード制御装置205に対するアドレス・バス・トラフィックを低減することである。最初の場合では、A、A+1などのラインは、プリフェッチ以前にキャッシュ202内に存在せず、従って通常、キャッシュ・ラインA+5がL2キャッシュ203内に存在することは期待できない。

【0052】ステップ626及び627が上述の理由から結合されるとき、ステップ627の通知は、ステップ626のプリフェッチに追加される4ビットの制御ビットにより実現され得る。4ビットは、1ビットの有効プリフェッチと、2ビットのストリーム識別と、1ビットのプリフェッチ方向を含む。キャッシュ・ラインA+5のアドレス及びこれらのビットから、ノード制御装置205は、キャッシュ・ラインA+6及びA+7に対するメモリ要求を生成する。前述のように、ノード制御装置205は任意の数のキャッシュ・ラインをプリフェッチするように実現され得る。ステップ628では、ノード制御装置205はキャッシュ・ラインA+6をプリフェッチ・バッファPBFR2 405にプリフェッチし、キャッシュ・ラインA+7を、プリフェッチ・バッファPBFR3 406にプリフェッチする。

【0053】ステップ628とステップ629との間の

破線矢印は、キャッシュ・ラインA+4に対するCPU201からの要求が、ステップ628の直後に必ずしも発生しないことを示す。

【0054】ステップ629では、CPU201によるキャッシュ・ラインA+4要求に対して、L1キャッシュ202がアクセスされる。キャッシュ・ラインA+4はステップ624でL1キャッシュ202に挿入されているので、キャッシュ・ラインA+4がCPU201に返却される。ステップ630では、ストリーム・アドレスが増分され、アドレスA+5により先導される。ステップ631では、キャッシュ・ラインA+5がL1キャッシュ202内に存在するか否かがチェックされ、存在しない場合、キャッシュ・ラインA+5がL2キャッシュ203またはバッファ402から、L1キャッシュ202にフェッチされる。

【0055】その後、ステップ632で、キャッシュ・ラインA+6がPBFR2 405からPBFR402に転送される。ステップ633では、キャッシュ・ラインA+7がPBFR3 406からPBFR2 405に転送される。その後、ステップ634で、ノード制御装置205が、キャッシュ・ラインA+8をプリフェッチするように通知される。本技法では、ステップ632におけるキャッシュ・ラインA+6のフェッチは、ノード制御装置205に、キャッシュ・ラインA+8をプリフェッチするように通知する(ステップ634)。ステップ635で、ノード制御装置205は、キャッシュ・ラインA+8をメモリ209からPBFR3 406にプリフェッチする。

【0056】その後、CPU201がキャッシュ・ラインを順次増分しながらアクセスし続ける限り(すなわち、CPU201が割当てられたストリーム内のキャッシュ・ラインをアクセスし続ける)、プロセスは増分方式で(ステップ636)、ステップ629乃至636をループし続ける。

【0057】上述の議論では、バス・インタフェース(BIU)401は、システム・メモリ209からキャッシュ・ラインのフェッチを実行する。

【0058】ノード制御装置205はスイッチ207上のポートであり得る。

【0059】有効アドレスはページ境界に跨り連続的であり、実アドレスは連続的でないので、ストリーム・アドレス・バッファ501内で2つのアドレスを比較するとき、有効アドレスを使用することがしばしば有利である。更に、増分アドレスを生成するために、カウンタが使用されてもよい。

【0060】上述のように、BIU401からメモリ209へのフェッチにおいて、制御ビットが使用され、ノード制御装置205に、キャッシュ・ラインをPBFR2405及びPBFR3 406にプリフェッチするように通知する。1ビットがノード制御装置205に、こ

の特定のライン要求が、ノード制御装置205によるそのバッファへのプリフェッチの実行を要求する旨を通知し得る。別の2ビットは、ノード制御装置に、プリフェッチに関連付けられるストリーム番号を通知し得る。別の1ビットは、アドレスが進行するキャッシュ・ラインの方向を示し得る。ノード制御装置205はプリフェッチを実行するように通知されるとき、こうしたプリフェッチを、CPU201の動作とは独立に実行し得る。

【0061】L1キャッシュ202及びL2キャッシュ203を含む場合、前記プロシジヤにおいて、キャッシュ・ラインがPBFR402からL1キャッシュ202に転送されるとき、同一のキャッシュ・ラインがL2キャッシュ203内にも含まれることになる。

【0062】L1キャッシュ202内にストリーム・バッファ・ラインの1つを有する利点は、L1キャッシュ202内のそのバッファ・ライン内に含まれる特定のキャッシュ・ラインが、プロセッサ201により要求されるときに、L1キャッシュ202内でミスではなく、ヒットが発生することである。技術的には、たとえ要求されるキャッシュ・ラインが、L1キャッシュ202に接続される別のバッファ内に含まれていても、ミスは発生する。こうしたミスにより、そのキャッシュ・ラインをそのストリーム・バッファ・ラインからCPU201に取り出すために、余分なハードウェア及びサイクル時間が要求される。論理的には、ストリーム・バッファ・キャッシュ・ラインの1つとして作用するL1キャッシュ202内のキャッシュ・ラインは、プリフェッチ・ストリーム・バッファ内に含まれると称される。

【0063】まとめとして、本発明の構成に関して以下の事項を開示する。

【0064】(1) プロセッサと、前記プロセッサにバスを介して接続されるシステム・メモリと、前記プロセッサに接続される第1のキャッシュと、前記プロセッサに接続される第2のキャッシュと、前記システム・メモリからプリフェッチされる1つ以上のデータ・ラインを記憶するストリーム・バッファ回路と、プリフェッチ状態を示すストリーム・フィルタ回路と、前記ストリーム・フィルタ回路に接続され、前記システム・メモリから前記第1及び第2のキャッシュ、並びに前記ストリーム・バッファ回路へのデータのフェッチ及びプリフェッチを選択的に制御する制御回路と、を含む、データ処理システム。

(2) 前記制御回路が、第1のキャッシュ・ラインに対する要求を前記プロセッサから受信する回路と、前記第1のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断する回路と、前記第1のキャッシュ・ラインが前記第1のキャッシュ内に存在せず、前記第1のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第1のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチする

回路と、前記第1のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第1のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにフェッチする回路と、前記第1のキャッシュ・ラインのアドレスを増分し、第1の増分アドレスを生成する回路と、前記増分アドレスを前記ストリーム・フィルタに記憶する回路と、を含む、前記(1)記載のシステム。

(3) 前記制御回路が、前記増分アドレスを有する第2のキャッシュ・ラインに対する要求を前記プロセッサから受信する回路と、前記第2のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断する回路と、前記増分アドレスが前記ストリーム・フィルタ内に存在するか否かを判断する回路と、前記増分アドレスを増分し、第2の増分アドレスを生成する回路と、前記第2の増分アドレスで開始するストリームを割当て回路と、前記第2の増分アドレスを有する第3のキャッシュ・ラインが、前記第1のキャッシュ内に存在するか否かを判断する回路と、前記第2のキャッシュ・ラインが前記第2のキャッシュ内に存在し、前記第2のキャッシュ・ラインが前記第1のキャッシュ内に存在しない場合、前記第2のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチする回路と、前記第2のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第2のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにフェッチする回路と、前記第3のキャッシュ・ラインが前記第2のキャッシュ内に存在し、前記第3のキャッシュ・ラインが前記第1のキャッシュ内に存在しない場合、前記第3のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにプリフェッチする回路と、前記第3のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第3のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにプリフェッチする回路と、を含む、前記(2)記載のシステム。

(4) 前記第2の増分アドレスを増分し、第3の増分アドレスを生成する回路と、前記第3の増分アドレスを有する第4のキャッシュ・ラインが、前記第2のキャッシュ内に存在するか否かを判断する回路と、前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第4のキャッシュ・ラインを前記システム・メモリから前記ストリーム・バッファ回路にプリフェッチする回路と、前記第3のキャッシュ・ラインに対する要求を前記プロセッサから受信する回路と、前記第3のキャッシュ・ラインを前記第1のキャッシュから前記プロセッサに送信する回路と、前記第4のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断する回路と、前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第4のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャ

ッシュにフェッチする回路と、前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第4のキャッシュ・ラインを前記ストリーム・バッファ回路から前記第1のキャッシュにフェッチする回路と、前記第3の増分アドレスを増分し、第4の増分アドレスを生成する回路と、前記第4の増分アドレスを有する第5のキャッシュ・ラインが、前記第2のキャッシュ内に存在するか否かを判断する回路と、前記第5のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第5のキャッシュ・ラインを前記第2のキャッシュから前記ストリーム・バッファ回路にフェッチする回路と、前記第5のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第5のキャッシュ・ラインを前記システム・メモリから前記ストリーム・バッファ回路にプリフェッチする回路と、を含む、前記 (3) 記載のシステム。

(5) 前記ストリーム・バッファ回路に記憶される1つ以上のキャッシュ・ラインが、前記第1のキャッシュ内に配置される、前記 (1) 記載のシステム。

(6) 前記ストリーム・バッファ回路に記憶される1つ以上のキャッシュ・ラインが、前記プロセッサを含むチップ内に配置される、前記 (1) 記載のシステム。

(7) 前記ストリーム・バッファ回路に記憶される1つ以上のキャッシュ・ラインが、前記プロセッサ及び前記システム・メモリに接続されるノード制御装置内に配置される、前記 (1) 記載のシステム。

(8) 前記第1のキャッシュが前記プロセッサと同一チップ上に配置される1次キャッシュであり、前記第2のキャッシュが前記チップの外部に配置される2次キャッシュである、前記 (1) 記載のシステム。

(9) 前記ストリーム・フィルタ回路が複数のストリームを追跡可能であり、前記ストリーム・フィルタ回路内の各エントリが、前記複数のストリームの1つを追跡し、前記各エントリが、当該エントリにより追跡される前記ストリームの妥当性を示す第1の標識と、当該エントリのアドレスの増分方向を示す第2の標識とを含む、前記 (1) 記載のシステム。

(10) 前記ストリーム・バッファ回路が各エントリに対して、1) ページ・アドレスと、2) ライン・アドレスと、3) 妥当性標識とを含む、前記 (1) 記載のシステム。

(11) 前記ストリーム・フィルタ回路内のエントリが実アドレスを有し、前記ストリーム・バッファ回路内のエントリが有効アドレスを有する、前記 (1) 記載のシステム。

(12) 前記プロセッサ、前記制御回路、前記ストリーム・フィルタ回路、前記第1のキャッシュ、及び前記ストリーム・バッファ回路の一部が、同一チップ上に配置される、前記 (1) 記載のシステム。

(13) 第2のプロセッサと、前記第2のプロセッサに

接続される前記システム・メモリと、前記第2のプロセッサに接続される第3のキャッシュと、前記第2のプロセッサに接続される第4のキャッシュと、前記第2のプロセッサに接続され、前記システム・メモリからプリフェッチされる1つ以上のデータ・ラインを記憶する第2のストリーム・バッファ回路と、前記第2のプロセッサに接続され、プリフェッチ状態を示す第2のストリーム・フィルタ回路と、前記第2のストリーム・フィルタ回路に接続され、前記システム・メモリから前記第3及び第4のキャッシュ、並びに前記第2のストリーム・バッファ回路へのデータのフェッチ及びプリフェッチを選択的に制御する、第2の制御回路と、を含む、前記 (1) 記載のシステム。

(14) データ処理システムにおいて、第1のキャッシュ・ラインに対する要求を前記プロセッサから受信するステップと、前記第1のキャッシュ・ラインが第1のキャッシュ内に存在するか否かを判断するステップと、前記第1のキャッシュ・ラインが前記第1のキャッシュ内に存在せず、前記第1のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第1のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチするステップと、前記第1のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第1のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにフェッチするステップと、前記第1のキャッシュ・ラインのアドレスを増分し、第1の増分アドレスを生成するステップと、前記増分アドレスをストリーム・フィルタに記憶するステップと、を含む、方法。

(15) 前記増分アドレスを有する第2のキャッシュ・ラインに対する要求を前記プロセッサから受信するステップと、前記第2のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断するステップと、前記増分アドレスが前記ストリーム・フィルタ内に存在するか否かを判断するステップと、前記増分アドレスを増分し、第2の増分アドレスを生成するステップと、前記第2の増分アドレスで開始するストリームを割当てるステップと、前記第2の増分アドレスを有する第3のキャッシュ・ラインが、前記第1のキャッシュ内に存在するか否かを判断するステップと、前記第2のキャッシュ・ラインが前記第2のキャッシュ内に存在し、前記第2のキャッシュ・ラインが前記第1のキャッシュ内に存在しない場合、前記第2のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチするステップと、前記第2のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第2のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにフェッチするステップと、前記第3のキャッシュ・ラインが前記第2のキャッシュ内に存在し、前記第3のキャッシュ・ラインが前記第1のキャッシュ内に存在

しない場合、前記第3のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにプリフェッチするステップと、前記第3のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第3のキャッシュ・ラインを前記システム・メモリから前記第1のキャッシュにプリフェッチするステップと、を含む、前記(14)記載の方法。

(16) 前記第2の増分アドレスを増分し、第3の増分アドレスを生成するステップと、前記第3の増分アドレスを有する第4のキャッシュ・ラインが、前記第2のキャッシュ内に存在するか否かを判断するステップと、前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第4のキャッシュ・ラインを前記システム・メモリから前記ストリーム・バッファ回路にプリフェッチするステップと、前記第3のキャッシュ・ラインに対する要求を前記プロセッサから受信するステップと、前記第3のキャッシュ・ラインを前記第1のキャッシュから前記プロセッサに送信するステップと、前記第4のキャッシュ・ラインが前記第1のキャッシュ内に存在するか否かを判断するステップと、前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第4のキャッシュ・ラインを前記第2のキャッシュから前記第1のキャッシュにフェッチするステップと、前記第4のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第4のキャッシュ・ラインを前記ストリーム・バッファ回路から前記第1のキャッシュにフェッチするステップと、前記第3の増分アドレスを増分し、第4の増分アドレスを生成するステップと、前記第4の増分アドレスを有する第5のキャッシュ・ラインが、前記第2のキャッシュ内に存在するか否かを判断するステップと、前記第5のキャッシュ・ラインが前記第2のキャッシュ内に存在する場合、前記第5のキャッシュ・ラインを前記第2のキャッシュから前記ストリーム・バッファ回路にフェッチするステップと、前記第5のキャッシュ・ラインが前記第2のキャッシュ内に存在しない場合、前記第5のキャッシュ・ラインを前記システム・メモリから前記ストリーム・バッファ回路にプリフェッチするステップと、を含む、前記(15)記載の方法。

(17) 前記ストリーム・バッファ回路に記憶される1つ以上のキャッシュ・ラインが、前記第1のキャッシュ内に配置される、前記(16)記載の方法。

(18) 前記第1のキャッシュが前記プロセッサと同一チップ上に配置される1次キャッシュであり、前記第2のキャッシュが前記チップの外部に配置される2次キャッシュである、前記(17)記載の方法。

(19) 前記ストリーム・フィルタ回路が複数のストリームを追跡可能であり、前記ストリーム・フィルタ回路内の各エントリが、前記複数のストリームの1つを追跡し、前記各エントリが、当該エントリにより追跡される前記ストリームの妥当性を示す第1の標識と、当該エントリのアドレスの増分方向を示す第2の標識とを含む、前記(18)記載の方法。

(20) 前記ストリーム・フィルタ回路内のエントリが実アドレスを有し、前記ストリーム・バッファ回路内のエントリが有効アドレスを有する、前記(19)記載の方法。

【図面の簡単な説明】

【図1】本発明により構成可能なマルチプロセッサ・システムを示す図である。

【図2】本発明により構成可能なデータ処理システムを示す図である。

【図3】ストリーム・フィルタ及びストリーム・バッファを示す図である。

【図4】図2に示されるシステムの詳細を示す図である。

【図5】本発明による機能図である。

【図6】本発明のフロー図である。

【図7】本発明のフロー図である。

【図8】本発明のフロー図である。

【図9】本発明のフロー図である。

【符号の説明】

100、200 データ処理システム

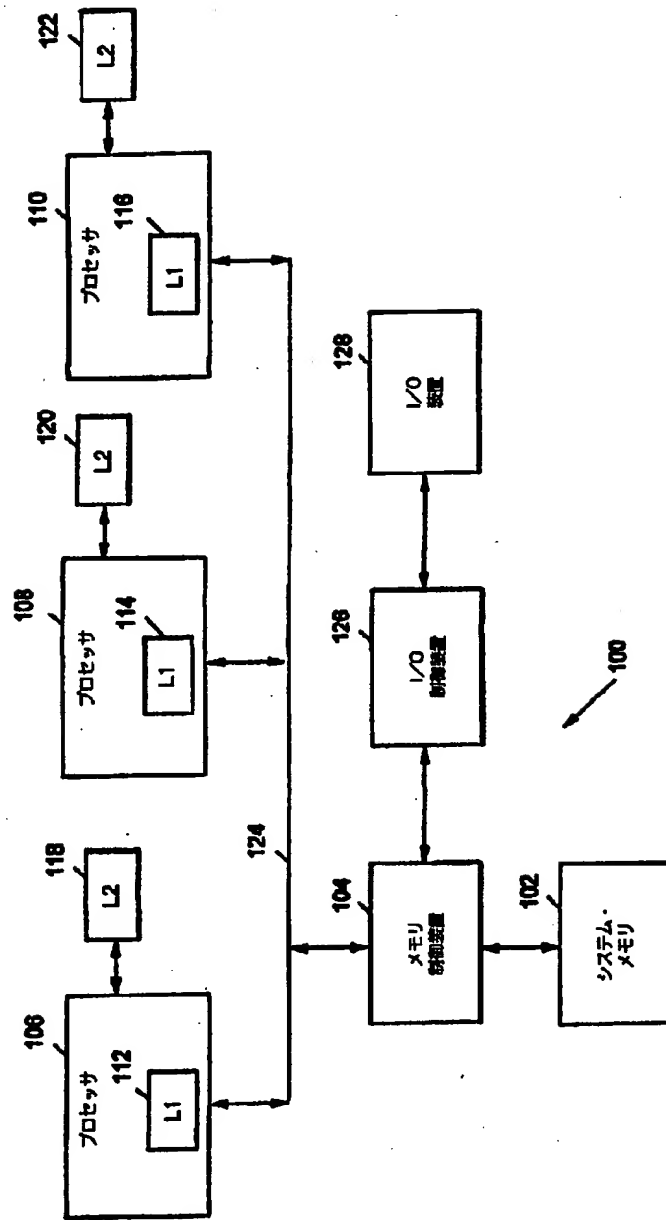
202 L1キャッシュ

203 L2キャッシュ

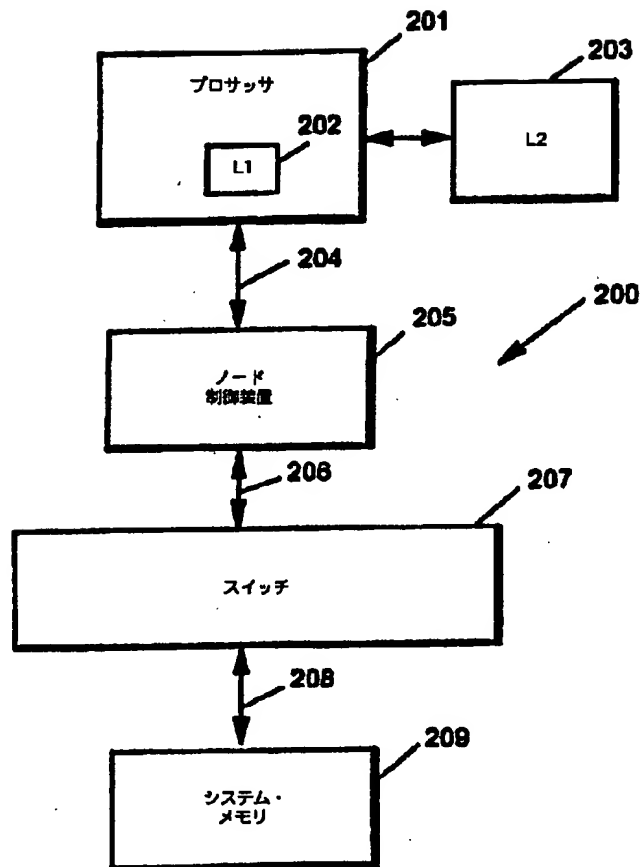
204、206、208 バス

402 プリフェッチ・バッファ

【図1】

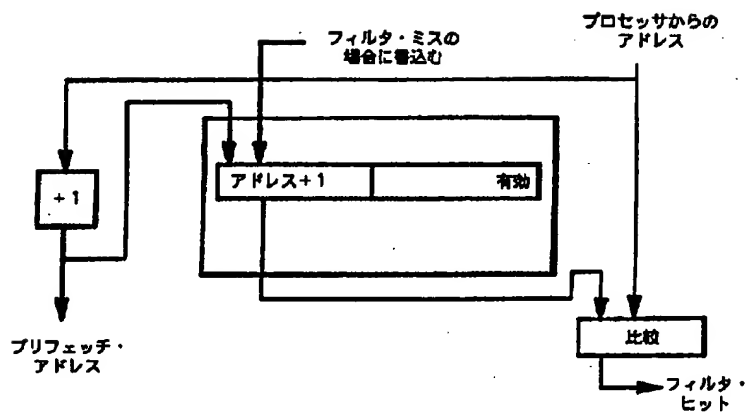


【図2】

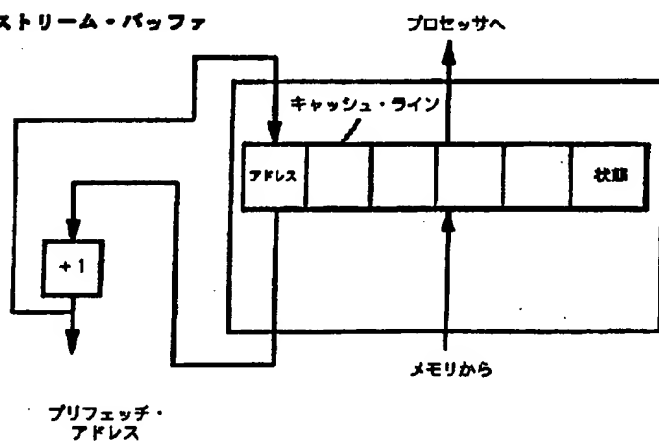


【図3】

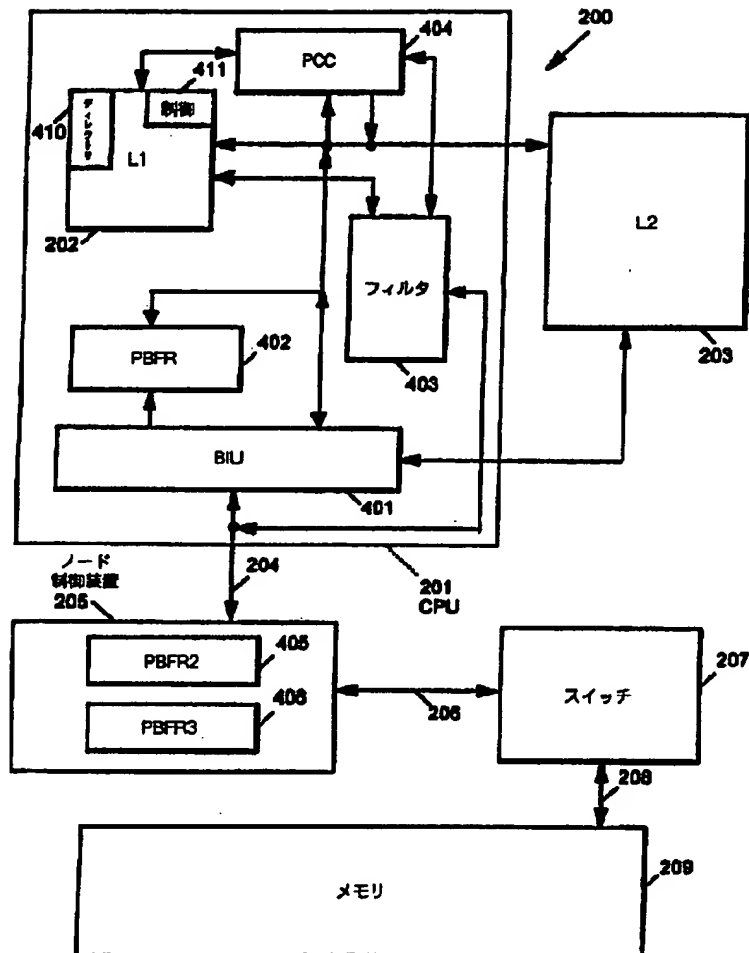
ストリーム・フィルタ



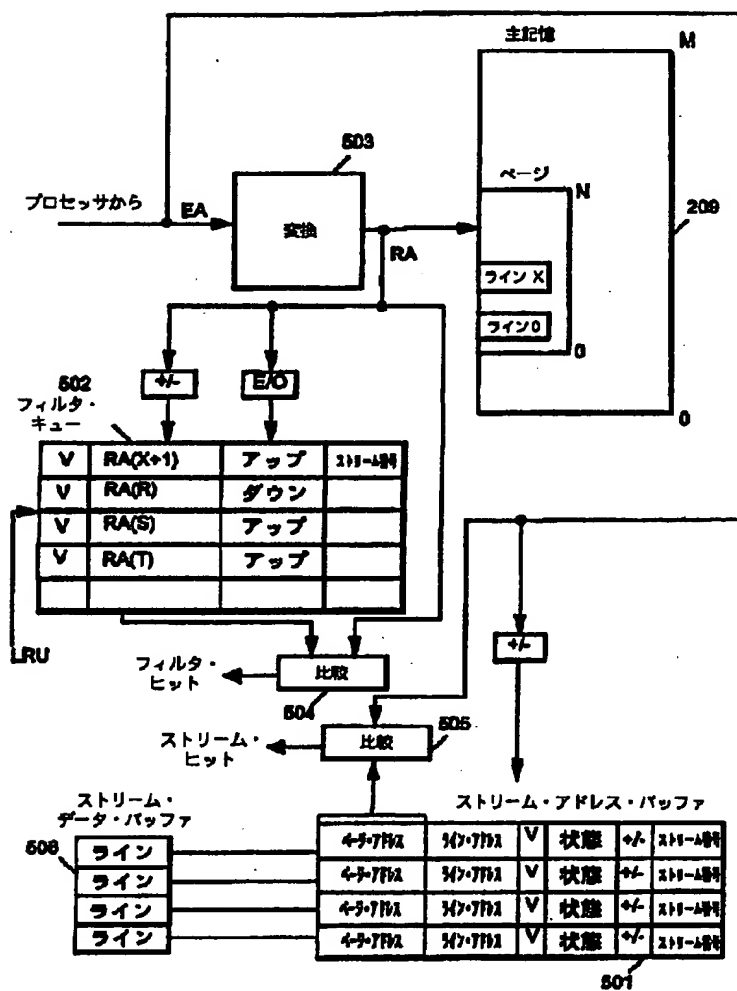
ストリーム・バッファ



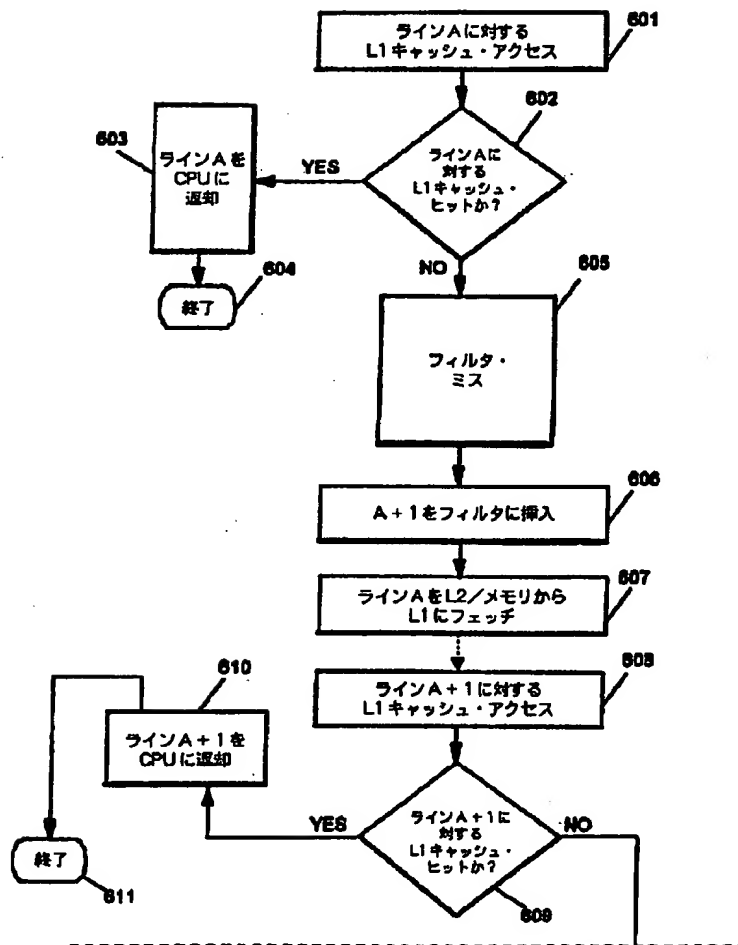
【図4】



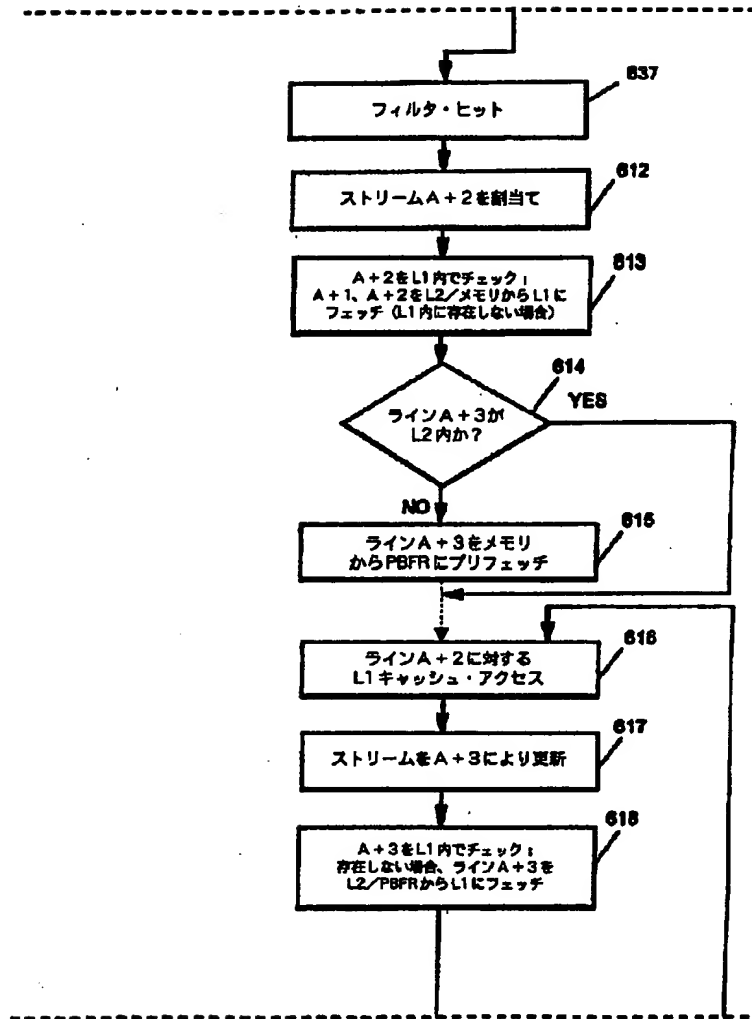
【図5】



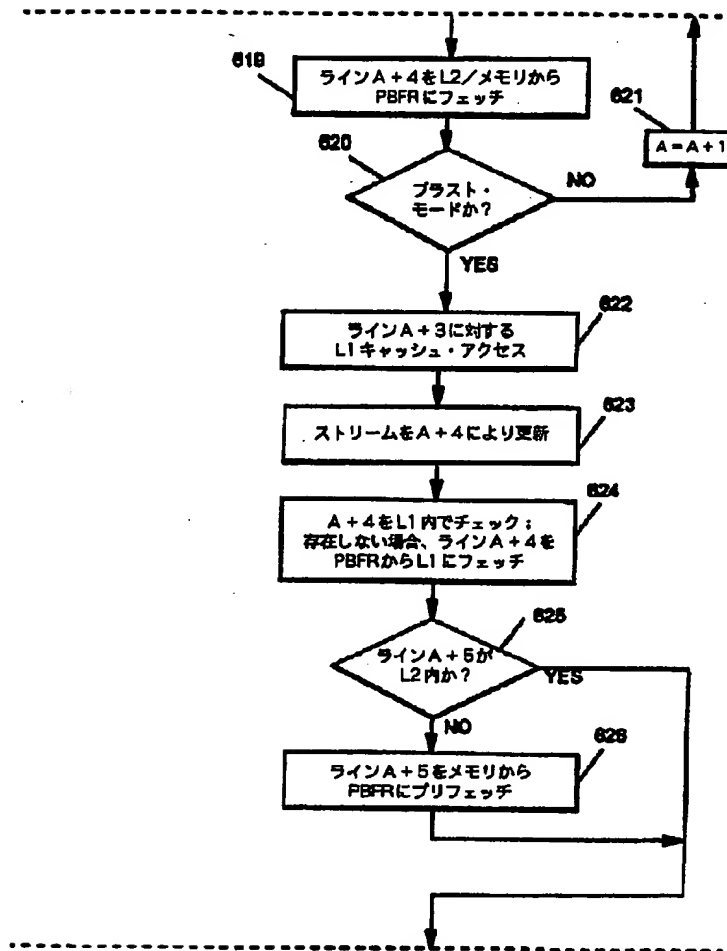
【図6】



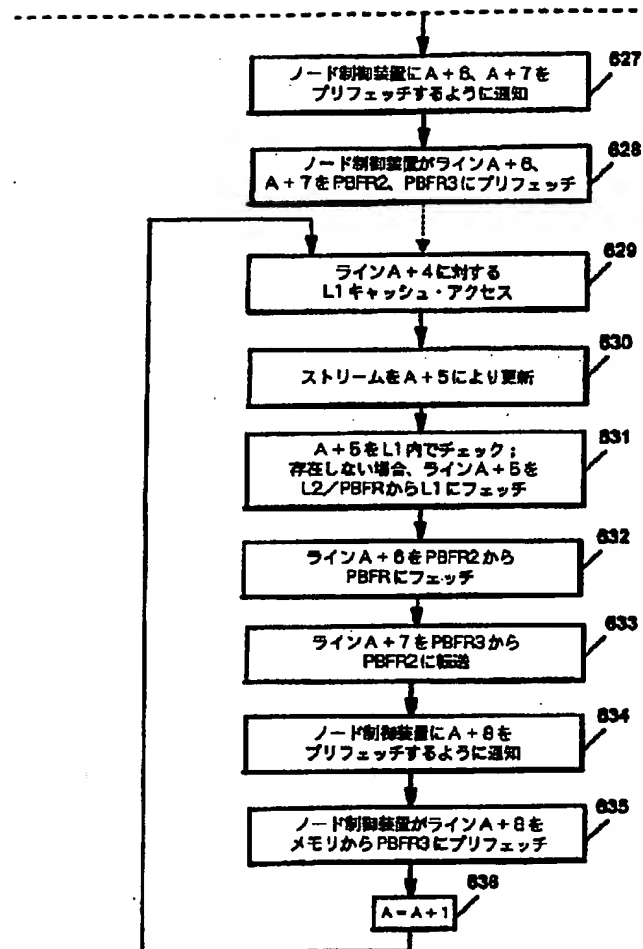
【図7】



【図8】



【図9】



フロントページの続き

(72)発明者 ドウワイン・アラン・ヒクス
アメリカ合衆国78660、テキサス州ブフル
ガービル、デューンズ・ドライブ 2405

(72)発明者 デビッド・スコット・レイ
アメリカ合衆国78728、テキサス州ジョー
ジタウン、ヤング・ランチ・ロード 700
(72)発明者 シーシン・ステファン・タン
アメリカ合衆国78726、テキサス州オー
スティン、バーブルック・ドライブ 9923

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.